Problem Statement

Does the rank of an author's books have anything to do with the number of pages, polarity of decriptions and titles, and ratings-count?

```
ranking_author["num_pages"].plot.bar(color="green")
plt.xlabel("Authors with Average Rank of 5")
plt.ylabel("Number of Pages")
plt.title("Bar Graph of Authors with an Average Rank of 5 Compared to the Number of Pages")
```

Out[37]: Text(0.5, 1.0, 'Bar Graph of Authors with an Average Rank of 5 Compared to the Number of Pages')

Bar Graph of Authors with an Average Rank of 5 Compared to the Number of Pages
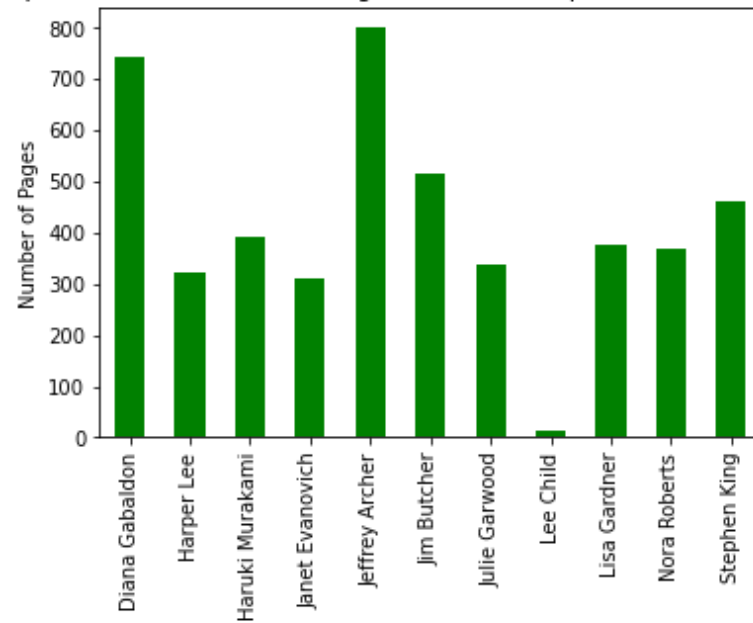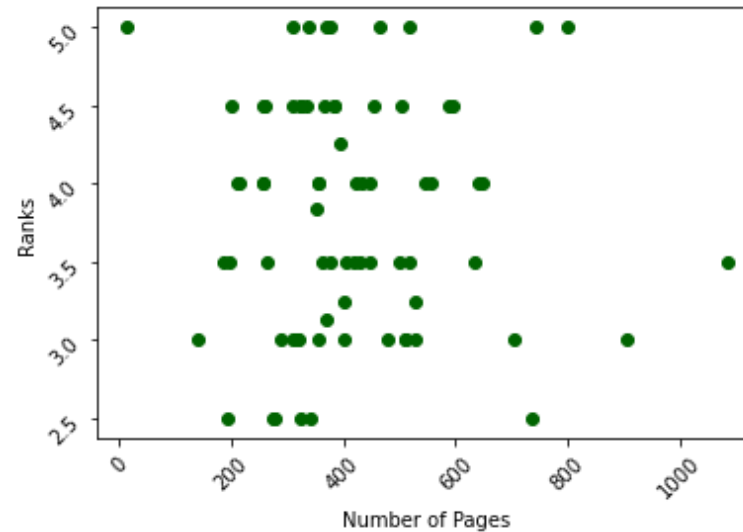
```
In [50]:  y = [385, 387, 389, 391, 393, 395, 397, 399, 401, 403, 405, 407, 409, 411]
          x = ["Rank 1", "Rank 2", "Rank 3", "Rank 4", "Rank 5"]


          plt.scatter(ranking_num.index, ranking_num, c="darkgreen", marker = "o")
          plt.yticks(rotation = 45)
          plt.xticks(rotation =45)


          plt.xlim = [0, 1000]
          plt.xlabel("Number of Pages")
          plt.ylabel("Ranks")
```
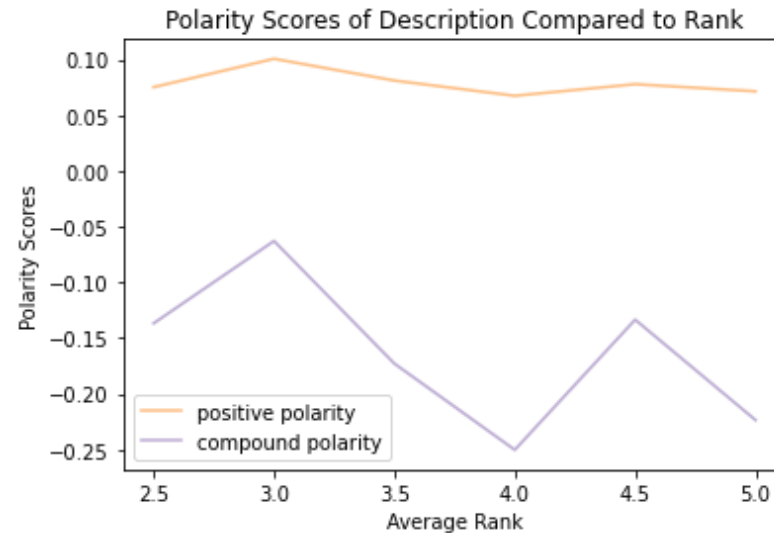
Out[50]: Text(0, 0.5, 'Ranks')

```python
color_map = {"polarity_positive":"#fdc086","polarity_compound":"#beaed4"}
label_map = {"polarity_positive":"positive polarity","polarity_compound":"compound polarity"}
fig, ax =plt.subplots()
grouped_pos = merged_data_titles.groupby("average_rank")["polarity_positive"].mean()
grouped_pos.plot(c=color_map["polarity_positive"], label=label_map["polarity_positive"])


grouped_neg = merged_data_titles.groupby("average_rank")["polarity_compound"].mean()
grouped_neg.plot(c=color_map["polarity_compound"], label=label_map["polarity_compound"])
plt.legend(labels = [label_map[x] for x in label_map])
plt.xlabel("Average Rank")
plt.ylabel("Polarity Scores")
plt.title("Polarity Scores of Description Compared to Rank")
```

Out[67]:   Text(0.5, 1.0, 'Polarity Scores of Description Compared to Rank')

```
In [68]:  color_map = {"title_x_polarity_positive":"#fdc086","title_x_polarity_compound":"#beaed4"}
          label_map = {"pos":"positive polarity", "compound":"compound polarity"}
          fig, ax =plt.subplots()
          grouped_pos = merged_data_titles.groupby("average_rank")["title_x_polarity_positive"].mean()
          grouped_pos.plot(c=color_map["title_x_polarity_positive"], label=label_map["pos"])
          grouped_compound = merged_data_titles.groupby("average_rank")["title_x_polarity_compound"].mean()

          grouped_compound.plot(c=color_map["title_x_polarity_compound"], label=label_map["compound"])
          plt.legend(labels = [label_map[x] for x in label_map])
          plt.xlabel("Average Rank")
          plt.ylabel("Polarity Scores")
          plt.title("Polarity Scores of Titles Compared to Rank")
```

Out[68]:   Text(0.5, 1.0, 'Polarity Scores of Titles Compared to Rank')
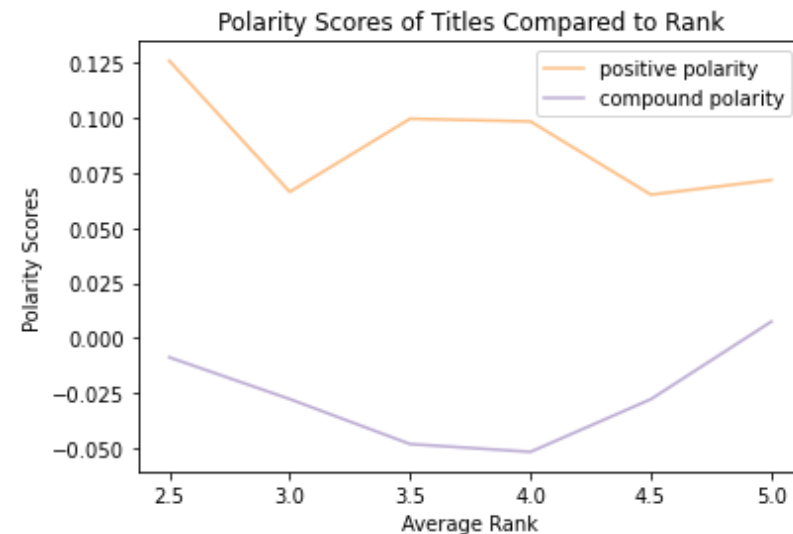
```
In [63]: ranking_rating.plot.bar(color="blue")

         plt.xlabel("Average Rank")
         plt.ylabel("Ratings Count")
         plt.title("Correlation of Average Rank to Ratings Count")
```

Out[63]: Text(0.5, 1.0, 'Correlation of Average Rank to Ratings Count')